

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 06-332716
(43)Date of publication of application : 02.12.1994

(51)Int.Cl. G06F 9/445
G06F 9/06
G06F 13/00
G06F 15/16

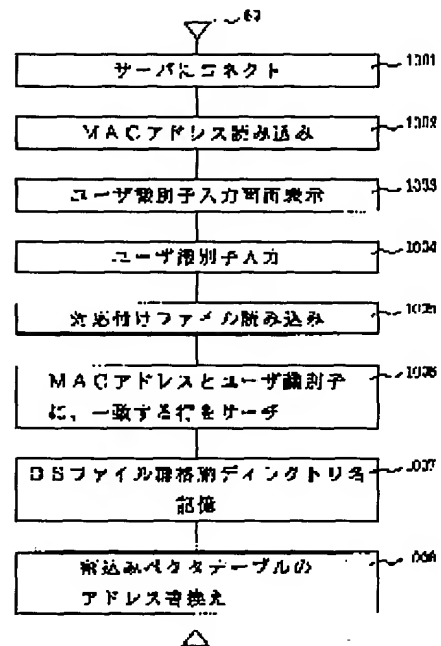
(21)Application number : 05-123794 (71)Applicant : HITACHI LTD
(22)Date of filing : 26.05.1993 (72)Inventor : TAKAHASHI YUMIKO
ITO HIROMICHI
NAKANE KEIICHI

(54) REMOTE BOOT SYSTEM

(57)Abstract:

PURPOSE: To improve an operation efficiency by operating a remote boot in the desired OS or environment setting of each user even when plural users use one client.

CONSTITUTION: Plural OS file groups whose kinds are different on the hard disk of a server are selected by using corresponding identifiers, and read in the main storage of the client. For example, the OS files are stored in server directories different for each kind, and a corresponding file in which the correspondence of the directories to the user identifiers is described is prepared. An LAN board BIOS initialization program is equipped with a user identifier inputting means 1004, and means 1006 and 1007 which obtain and store the name of the directory in which the OS file corresponding to the user identifier is stored from the corresponding file, and the OS files are read from the directory.



LEGAL STATUS

[Date of request for examination]
[Date of sending the examiner's decision of rejection]
[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]
[Date of final disposal for application]
[Patent number]
[Date of registration]
[Number of appeal against examiner's decision of rejection]
[Date of requesting appeal against examiner's decision of rejection]
[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

*** NOTICES ***

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. **** shows the word which can not be translated.
3. In the drawings, any words are not translated.

CLAIMS

[Claim(s)]

[Claim 1] The remote boot system characterized by for the aforementioned client to read into the primary storage of this client after selection two or more operating system file groups from which the kind on the hard disk of the aforementioned server differs using an identifier in the remote boot which reads into the primary storage of a client the operating system file group stored on the hard disk of the aforementioned server via a network by the system which connected the information processor called the information processor called client and server in the network.

[Claim 2] The remote boot system are a remote boot system according to claim 1, and carry out having an identifier acquisition means to prepare the matching file which matches the operating system file group and the aforementioned identifier of the aforementioned plurality, and to acquire the aforementioned identifier on the hard disk of a server, and the aforementioned identifier and the means which reads the desired operating system file group for remote boot using the aforementioned matching file as the feature.

[Claim 3] The remote boot system characterized by being a remote boot system according to claim 1 or 2, and the identifier for choosing an operating system file group being a user-identification child who discriminates each user.

[Claim 4] The remote boot system characterized by being a remote boot system according to claim 1 or 2, and the identifier for choosing an operating system file group being an OS identifier which shows the kind of OS.

[Claim 5] The remote boot system characterized by being a remote boot

system according to claim 1 or 2, and the identifier for choosing an operating system file group being an environmental identifier which shows an environmental file type.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Industrial Application] Information processors, such as a personal computer, and the information processing network which connected two or more these are started, and it is related with the program for loading the operating system of an information processor from other information processors especially.

[0002]

[Description of the Prior Art] In response to the influence of downsizing of a computer, development of a Network Operating System, etc., the system connected in networks, such as LAN (Local Area Network), is increasing information processors, such as a personal computer (it omits Following PC), by in recent years. These systems consist of an information processor called client and an information processor called server in many cases. A client can use the printer which accessed the file on the hard disk (it omits Following HD) of a server, or was connected to the server through the network. Such a system is called client/server system.

[0003] By the way, in information processors, such as PC, it stores in nonvolatile storages, such as a floppy disk (it omits Following FD) by which this information processor possesses the operating system (it omits Following OS), or HD, and at the time of starting of this information processor, this OS is read on the primary storage of this information processor, and OS is started. This OS starting operation is called boot of OS.

[0004] In the above-mentioned client/server system, the program file of OS is stored on HD of a server, this OS program file is read from HD of a server on the primary storage of a client via a network at the time of the system startup of a client, and booting OS is also performed. It is called remote boot, and HD of each client can be made unnecessary, and access speed does not

need to use low speed FD, and such a boot method of OS has the said advantage which can perform unitary management of OS program file further.

[0005] In the above-mentioned remote boot, when the models of each client differ, OS program file for every model is prepared on Server HD, and it is necessary to read OS program file suitable for the model of client at the time of remote boot execution.

[0006] In this case, matching with two or more OS files placed on HD of a server and a client model was performed using the MAC (Medium Access Control) address of the LAN board with which this server and this client are equipped conventionally. Here, in case a Media Access Control Address delivers and receives the communication data in LAN, it is used as the address of the LAN board of each information processor, and the peculiar value is assigned to this one one-sheet board.

[0007] The above-mentioned matching is explained concretely below.

[0008] The system configuration at the time of remote boot execution is shown in drawing 5. A client 2, a client 3, a client 4, and a client 5 access the file 8 on HD7 of a server 1 in common via a network 6. A client 2 and a client 3 are X type models, and a client 4 and a client 5 are Y type models.

Moreover, the Media Access Control Addresses of clients 2, 3, 4, and 5 are 5A003B77H (H shows a hexadecimal), 4B931211H, 2C984A67H, and 75682732H, respectively.

[0009] The file organization required for it in case each aforementioned client performs remote boot to drawing 8 on a server HD 7 is shown. IPL (Initial Program Loader) 21 and 31 is a program for loading OS. DOS.SYS 22 and 32 and INOUT.SYS 23 and 33 are the kernel programs of OS. Among these, DOS.SYS 22 and 32 has the basic function of OS's, such as file management, memory management, and process control, and INOUT.SYS 23 and 33 controls a peripheral device. COM.COMs 24 and 34 are command interpreters, analyze the inputted command and execute the specified command program. CONF.SYS 25 and 35 is a configuration file which specifies a device driver etc., in order that a user may add a peculiar function to OS. AUTO.BAT 26 and 36 is an automatic execution batch file which sets the program performed automatically as the during starting of OS. In addition, an above-mentioned file is named generically and it is called OS file group. OS file group stored under the directory XTYPE81 is an object for

models aforementioned X type, and OS file group stored under the directory YTYPE82 is an object for models aforementioned Y type. MATCH.SYS83 is a file for performing matching with the aforementioned Media Access Control Address and the aforementioned OS file group.

[0010] Drawing 9 is the content of aforementioned MATCH.SYS83 which is a text file. The 1st line and the 2nd line show that 5A003B77H which are the Media Access Control Address of a client 2 and a client 3, 4B931211H, and the aforementioned directory XTYPE81 are matched, and the 3rd line and the 4th line show that 2C984A67H which are the Media Access Control Address of a client 4 and a client 5, 75682732H, and the aforementioned directory YTYPE82 are matched.

[0011] Thus, a client distinguishes the directory where OS file group for these client models was stored from the Media Access Control Address of the LAN board of this client using aforementioned matching file MATCH.SYS83, and performs remote boot using OS file group under this directory. OS file group suitable for the client model can be chosen by this, and when two or more client models are intermingled, remote boot can be carried out correctly.

[0012] It is indicated by U.S. patent No.240,955 about remote boot.

[0013]

[Problem(s) to be Solved by the Invention] By the way, kinds, such as OS, a front-end processor, and a graphical user interface program, and the initial state after starting change with the work content for every user, or liking. Moreover, kinds, such as OS to use, a front-end processor, and a graphical user interface program, and the initial state after starting may change with work contents which the same user also performs.

[0014] However, in the conventional remote boot function, when two or more users used the same client, there was a problem that the content of an environmental setting which determines the kind of OS and the initial state after starting of OS for every user could not be chosen.

[0015] Moreover, in the conventional remote boot function, there was a problem that the content of an environmental setting to which a user decides the kind of OS and the initial state after OS starting at the time of OS starting could not be chosen.

[0016] Therefore, the purpose of this invention is about the kind of OS of the client used for every user, and the content of an environmental setting at the time of starting of this OS to offer a selectable remote boot system.

[0017]

[Means for Solving the Problem] In order to attain the above-mentioned purpose, this invention established OS file read-out means which reads desired OS file group out of two or more OS file groups which described OS file group corresponding to a user-identification child, and which matched and were placed on Server HD using the file, and a user-identification child acquisition means, the aforementioned user-identification child and the aforementioned matching file in the information processing system which consists of a server and a client.

[0018]

[Function] A user-identification child's input is urged to a user-identification child acquisition means by the screen display etc., and it acquires the user-identification child whom a user inputs from a keyboard etc. OS file readout means chooses desired OS file group out of two or more OS file groups which described matching with this user-identification child, a user-identification child, and OS file group and which matched and were placed on Server HD using the file, and reads it on a client. By this, OS file group beforehand specified for every user by the matching file can be used, and the kind of OS and an environmental setup at the time of starting of this OS can be chosen for every user.

[0019]

[Example] Hereafter, one example of this invention is explained using drawing. The system configuration in this example is the same as that of what was shown by drawing 5 in the conventional example.

[0020] The matching file which performs matching with a model and OS file group for every user required for starting of a client 2, a client 3, a client 4, and a client 5, and the model identifier of the aforementioned client 2, a client 3, a client 4, and a client 5 and a user-identification child and the aforementioned OS file group is stored in HD7 of a server 1.

[0021] Drawing 4 is the example of composition of the client 2 in one example of this invention. As for CPU and 42, 41 is [a primary storage and 43] ROMs. The BIOS program which controls an information processor is stored in ROM43. The display controller for the keyboard controller by which 44 controls the key input from a keyboard 45, HD controller by which 46 controls the R/W to HD47, FD controller which controls the R/W to FD by which the FD drive 49 is equipped with 48, and 50 controlling the indicative

data to a display 51, and 55 are buses which transmit the signal between CPU41, a primary storage 42, ROM43, and each controller. The LAN board 9 is the LAN controller 52 and the LAN board BIOS. It consists of ROM53 and Media Access Control Address ROM54. LAN board BIOS The program which controls remote boot is stored in ROM53.

[0022] Drawing 6 is drawing showing an example of the memory map of the aforementioned client 2. A system program 61, FD The BIOS program 62 and the display BIOS program 63 are stored in the above ROM 43. ROMs for inclusion peripheral devices which check hardware in a system program 61 as shown in drawing 6 (b), such as the initial diagnostic program 66 and FDD There are the BIOS program 67 and a boot program 68 which reads and performs IPL from storage. BIOS of the aforementioned LAN board 9 LAN board BIOS stored in ROM53 The ROM program 64 is arranged from C000:0000H street by this example. This LAN board BIOS It is a convention that an initializer 69 is written in the start-address portion of the ROM program 64 following a code, 55H [for example,], specific AAH, and the specific aforementioned code as shown in drawing 6 (c). Moreover, it is FD for transposing the I/O to FD served by interruption from address C000:1000H street to the I/O to a server HD 7. The BIOS emulator program 70 is stored. Moreover, the interrupt vector table 65 is put on the portion of a 0000:03FFH address from 0000:0000H street.

[0023] The content of matching file MATCH.SYS105 is shown in drawing 11 . MATCH.SYS105 is a file for making the identifier which doubled the Media Access Control Address which is a model identifier, and the user-identification child, and OS file group prepared for every aforementioned user correspond. In this example, SUZUKI and SATO are used as a user-identification child.

[0024] For example, it is shown that the 1st line makes XSUZUKI correspond as a directory where OS file group was stored from Media Access Control Address 5A003B77H and the user-identification child SUZUKI of a client 2, and the 2nd line makes XSATO correspond as a directory where OS file group was stored from Media Access Control Address 5A003B77H and the user-identification child SATO of a client 2. The matching same from the 3rd line to the 8th line is shown below.

[0025] In case each client performs remote boot in the system configuration of drawing 5 to drawing 10 , the required file organization on a server HD 7

is shown. XSUZUKI101, YSUZUKI102, XSATO103, and YSATO104 are the directories prepared for every client model and user, and OS file group is stored in each directory. Although each file which constitutes OS file group from below is explained, since it is the same as that of the explanation of a same name file described by the Prior art, it omits about the detailed functional description of each aforementioned file here.

[0026] IPL71, DOS.SYS72, INOUT.SYS73, and COM.COM74 are OS files peculiar to an X type model. CONF.SYS75 and AUTO.BAT76 are files by which a peculiar environmental setup in case SUZUKI performs remote boot from an X type model is made. Similarly, it is the file by which a peculiar environmental setup CONF.SYS75' and in case, as for AUTO.BAT76', SATO performs remote boot from an X type model is made. IPL81, DOS.SYS82, INOUT.SYS83, and COM.COM84 are OS files peculiar to a Y type model. Although above-mentioned OS file and an above-mentioned name are the same, the contents differ. CONF.SYS85 and AUTO.BAT86 are files by which a peculiar environmental setup in case SUZUKI performs remote boot from a Y type model is made. Similarly, it is the file by which a peculiar environmental setup CONF.SYS85' and in case, as for AUTO.BAT86', SATO performs remote boot from a Y type model is made.

[0027] Next, before explaining operation at the time of remote boot, operation when a client 2 performs boot from FD is explained below using the flow chart of drawing 2.

[0028] If powering on or reset directions starts by the user, the above-mentioned system program 61 will be performed. The initial diagnostic program 66 in a system program 61 performs an initial diagnosis of a memory check etc. in Step 201. Next, the aforementioned ROM for peripheral devices Control moves to the BIOS program 67 and processing from Step 202 to Step 206 is performed. At Step 202, the inclusion peripheral device of FDD49 grade and its controller are initialized. At this time, the entry address of the ROM program for serving the radial transfer to FDD49 grade is set as a primary storage 42 as an interrupt vector table. In case radial transfer to FDD49 grade is performed, the processing program specified to be the aforementioned interrupt vector table can be called by publishing a software interrupt from a program. At Step 203, the existence of BIOSROM of the add-in board with which the option slot of an information processor etc. was equipped is investigated. At this example, it is add-in

board BIOS. Since he has promised the data located in the start address of a ROM program as 55H and AAH, the address in which the data of 55H and AAH exist continuously is investigated. If add-in board BIOS is detected as a result of this investigation, in Step 204, the initializer of this add-in board BIOS will be performed. After the LAN board BIOS is detected at the time of the remote boot mentioned later, this initializer 69 is performed. Next, Steps 203-205 are repeated until it performs the end judging of add-in board BIOS investigation and ends by whether all the addresses were investigated in Step 206. When investigation of add-in board BIOS is completed, it moves to the above-mentioned boot program 68, and in Step 206, control reads IPL from FDD49 and performs it. Next, Above IPL reads DOS.SYS which is the kernel program of OS, and INOUT.SYS in Step 207, and performs.

Specifically, after INOUT.SYS initializes a peripheral device (the check of the connection state of a peripheral device, initialization of a driver routine), initialization routine of DOS.SYS which is the main part of OS is performed, and reservation of a working area, reservation of a buffer area, etc. are performed. Next, after building into a system driver programs, such as a peripheral device specified with reference to CONF.SYS in Step 208, COM.COM which is a command interpreter is read and performed at Step 209. COM.COM reads and performs AUTO.BAT which is an automatic execution batch file in Step 210.

[0029] As drawing 6 showed, the data stored in the start address of the LAN board BIOS program 64 which controls remote boot are 55H and AAH. Therefore, this LAN board BIOS is detected at the above-mentioned step 203. The initializer 69 of this LAN board BIOS is performed at Step 204.

[0030] Next, operation of the aforementioned initializer 69 is explained using the flow chart of drawing 1.

[0031] First, connection SHON with a server 1 is established through the LAN controller 5 at Step 1001. Next, Media Access Control Address ROM54 on the LAN board 9 is accessed at Step 1002, and a Media Access Control Address is read. Next, a user-identification child input screen is expressed on a display 51 as Step 1003. Next, if a user-identification child is inputted into a keyboard 45 at Step 1004, matching file MATCH.SYS105 on a server HD 7 will be read at Step 1005. the line which is in agreement with the aforementioned Media Access Control Address and the aforementioned user-identification child at Step 1006 -- this -- by searching out of matching

file MATCH.SYS105, the directory name in which OS file group corresponding to this Media Access Control Address and this user-identification child was stored is obtained, and this directory name is memorized at Step 1007

[0032] For example, a Media Access Control Address memorizes the directory of XSUZUKI of matching file MATCH.SYS105 which they showed by drawing 11 when 5A003B77H and the above-mentioned user-identification child were SUZUKI(s) from the 1st line. Next, it is FD about the interrupt vector table for serving the radial transfer to the above FDD49 at Step 1008. It rewrites to the address for calling the BIOS emulator program 70, and the LAN board BIOS initializer 69 is ended.

[0033] Drawing 7 is the memory map which expanded the interrupt vector table 65 shown by drawing 6 . In drawing 7 , the 0000:0064H street in an interrupt vector table 65 is a field where the entry address for calling the radial transfer service to above-mentioned FDD49 is stored. As Step 202 of drawing 2 shows to drawing 7 (a) as mentioned above, it is FD about the aforementioned entry address. It is set as the F000:0000H street in which the BIOS program 62 is stored. The initializer 69 of the above-mentioned LAN board BIOS program 64 is FD so that the aforementioned entry address may be described in drawing 7 (b) in Step 1008 of drawing 1 as mentioned above. It rewrites to the C000:0000 street in which the BIOS emulator program 70 is stored.

[0034] Next, the above FD Operation of the BIOS emulator program 70 is explained below using the flow chart of drawing 3 . FD The BIOS emulator program 70 reads OS file group storing directory name [finishing / storage] at the above-mentioned step 1007 in Step 301. Next, FD The aforementioned OS file group storing directory name is added to the file name passed as a parameter from the program which called the BIOS emulator program 70. Next, a file is read from a server HD 1 at Step 303 using the file name which performed the aforementioned addition.

[0035] For example, since a directory name called XSUZUKI was memorized when a Media Access Control Address was [a user-identification child] SUZUKI in 5A003B77H as mentioned above, when the file name "IPL" is read as a parameter and there is a demand, read-out is performed from a server HD 1 by file name called "XSUZUKI/IPL".

[0036] Above-mentioned interrupt vector rewriting and Above FD OS file

read from FD in Steps 206-210 of drawing 2 by the BIOS emulator program 70 is read out of OS file group stored under the directory prepared for every combination of the Media Access Control Address on a server HD 1, and a user-identification child.

[0037] By the above operation, OS can be risen using the kernel program of different IPL and different OS for every user, a command interpreter and an automatic execution batch file, a peripheral-device control file, etc. Thereby, when using one set of an information processor for a shift by two or more users, it is effective in the ability to offer the optimal environment for each user.

[0038] Although the aforementioned example showed the example which discriminates OS file group from a Media Access Control Address by the user-identification child, and boots OS, it is also possible to discriminate the model of client using the model code written to ROM43. Thus, as long as it can specify the model of client, you may use identifiers other than a Media Access Control Address. Moreover, when a client consists of only same models, you may use only a user-identification child. Furthermore, it is also possible to change to a user-identification child and to choose OS which carries out remote boot by using OS identifier which shows the kind of OS, and the environmental configuration file used at the time of remote boot can also be chosen by using the environmental identifier which shows the kind of environmental configuration file. Of course, it is also possible to use it combining each identifier.

[0039]

[Effect of the Invention] As explained above, when according to this invention the kind of OS risen by remote boot and environment can be set up for every user and it uses one set of an information processor for a shift by two or more users, it is effective in the ability to offer the optimal environment for each user.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

[Drawing 1] It is the flow chart which shows the content of execution of the

initializer of the LAN board BIOS in one example of this invention.

[Drawing 2] It is the flow chart which shows starting operation of the client in the example of 1 ***** of this invention.

[Drawing 3] FD in one example of this invention It is the flow chart which shows operation of a BIOS emulator program.

[Drawing 4] It is the block diagram of the client in one example of this invention.

[Drawing 5] It is drawing showing the composition of the whole system in the conventional example and one example of this invention.

[Drawing 6] It is drawing showing the memory map of the client in one example of this invention.

[Drawing 7] It is drawing showing the interrupt vector of the client in one example of this invention.

[Drawing 8] It is drawing showing OS file organization for remote boot on HD of the server in the conventional remote boot.

[Drawing 9] It is drawing showing the content of the matching file in the conventional remote boot.

[Drawing 10] It is drawing showing OS file organization for remote boot on HD of the server in one example of this invention.

[Drawing 11] It is drawing showing the content of the matching file in one example of this invention.

[Description of Notations]

1 -- Server

2-3-4-5 -- Client

6 -- Network

7 -- HD

8 -- File

9 -- LAN board,

41 -- CPU

42 -- Primary storage

43 -- ROM

44 -- Keyboard controller,

45 -- Keyboard

46 -- HD controller,

47 -- HD

48 -- FD controller,

49 -- FDD,
50 -- Display controller,
51 -- Display
52 -- LAN controller,
53 -- LAN board BIOS ROM
54 -- Media Access Control Address ROM,
61 -- System program
62 -- FD BIOS program,
63 -- Display BIOS program,
64 -- LAN board BIOS program,
65 -- Interrupt vector table,
66 -- Initial diagnostic program,
67 -- ROM for peripheral devices BIOS program,
68 -- Boot program
69 -- Initializer
70 -- FD BIOS emulator program,
81-82-101-102-103-104 -- OS file storing directory for remote boot,
83-105 -- The content of a matching file.

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平6-332716

(43) 公開日 平成6年(1994)12月2日

(51) Int.Cl. ⁵	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 9/445				
9/06	4 1 0 D	9367-5B		
13/00	3 5 7 Z	7368-5B		
15/16	4 2 0 S	9190-5L		
		9367-5B		
			G 0 6 F 9/ 06 4 2 0 J	
			審査請求 未請求 請求項の数 5 O L (全 10 頁)	

(21) 出願番号 特願平5-123794

(22) 出願日 平成5年(1993)5月26日

(71) 出願人 000005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(72) 発明者 高橋 由美子

神奈川県横浜市戸塚区吉田町292番地株式

会社日立製作所マイクロエレクトロニクス

機器開発研究所内

(72) 発明者 伊藤 浩道

神奈川県横浜市戸塚区吉田町292番地株式

会社日立製作所マイクロエレクトロニクス

機器開発研究所内

(74) 代理人 弁理士 小川 勝男

最終頁に続く

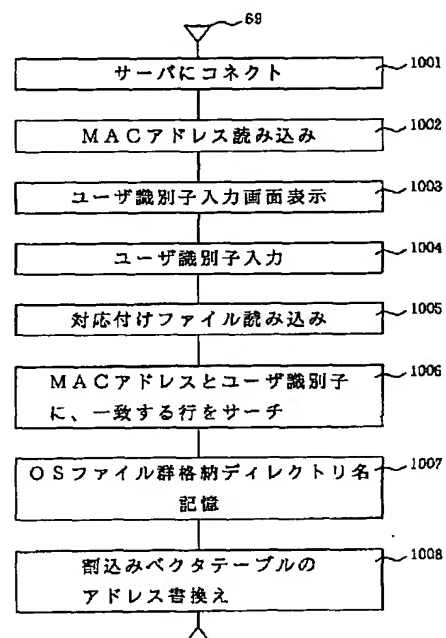
(54) 【発明の名称】 リモートブートシステム

(57) 【要約】 (修正有)

【構成】 サーバのハードディスク上の種類の異なる複数のOSファイル群を対応付けした識別子を用いて選択して、クライアントの主記憶に読み込む。例えば、OSファイルを、その種類毎に異なるサーバディレクトリに格納するとともに、ディレクトリとユーザ識別子との対応を記述した対応付けファイルを用意した。LANボードBIOS初期化プログラムに、ユーザ識別子入力手段1004と、対応付けファイルからユーザ識別子に対応したOSファイルを格納したディレクトリ名を取得し記憶する手段1006、1007を設け、ディレクトリからOSファイルを読み出す。

【効果】 1台のクライアントを複数のユーザが使用した場合でも、各ユーザの所望のOSあるいは環境設定でのリモートブートができ、作業効率が向上する。

図 1



【特許請求の範囲】

【請求項1】 クライアントと呼ばれる情報処理装置とサーバと呼ばれる情報処理装置をネットワークで接続したシステムで、前記クライアントが前記サーバのハードディスク上に格納したオペレーティングシステムファイル群を、ネットワーク経由でクライアントの主記憶に読み込みリモートブートにおいて、前記サーバのハードディスク上の種類の異なる複数のオペレーティングシステムファイル群を識別子を用いて選択後、該クライアントの主記憶に読み込むことを特徴とするリモートブートシステム。

【請求項2】 請求項1記載のリモートブートシステムであって、サーバのハードディスク上には、前記複数のオペレーティングシステムファイル群と前記識別子を対応付ける対応付けファイルを用意し、前記識別子を取得する識別子取得手段、前記識別子と前記対応付けファイルを用いて所望のリモートブート用オペレーティングシステムファイル群を読み出す手段とを持つことを特徴とするリモートブートシステム。

【請求項3】 請求項1又は請求項2記載のリモートブートシステムであって、オペレーティングシステムファイル群を選択するための識別子が、各ユーザを識別するユーザ識別子であることを特徴とするリモートブートシステム。

【請求項4】 請求項1又は請求項2記載のリモートブートシステムであって、オペレーティングシステムファイル群を選択するための識別子が、OSの種類を示すOS識別子であることを特徴とするリモートブートシステム。

【請求項5】 請求項1又は請求項2記載のリモートブートシステムであって、オペレーティングシステムファイル群を選択するための識別子が、環境ファイルの種類を示す環境識別子であることを特徴とするリモートブートシステム。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 パーソナルコンピュータ等の情報処理装置、及びこれを複数接続した情報処理ネットワークに係り、特に、情報処理装置のオペレーティングシステムを他の情報処理装置からロードするためのプログラムに関する。

【0002】

【従来の技術】 近年コンピュータのダウンサイジング化、ネットワークオペレーティングシステムの発達などの影響を受けて、パーソナルコンピュータ（以下PCと略す）等の情報処理装置をLAN（Local Area Network）等のネットワークで接続したシステムが増加している。これらのシステムは、クライアントと呼ばれる情報処理装置と、サーバと呼ばれる情報処理装置から構成されることが多い。クライアントは、ネ

ットワークを介しサーバのハードディスク（以下HDと略す）上のファイルをアクセスしたり、サーバに接続されたプリンタを利用することが可能である。このようなシステムを、クライアントサーバシステムという。

【0003】 ところで、PC等の情報処理装置においては、そのオペレーティングシステム（以下OSと略す）を該情報処理装置が具備するフロッピーディスク（以下FDと略す）、あるいはHD等の不揮発性記憶装置に格納し、該情報処理装置の立ち上げ時に、該OSを該情報処理装置の主記憶上に読み出し、OSを立ち上げるようになっている。このOS立ち上げ動作は、OSのブートと呼ばれている。

【0004】 上述のクライアントサーバシステムにおいては、サーバのHD上にOSのプログラムファイルを格納しておき、クライアントのシステム立ち上げ時に、サーバのHDから該OSプログラムファイルをネットワーク経由でクライアントの主記憶上に読み出し、OSをブートするといったことも行われている。このようなOSのブート方法は、リモートブートと呼ばれ、各クライアントのHDを不要に出来、またアクセス速度が低速なFDを使用しなくても良く、さらにOSプログラムファイルの一元管理が出来るといった利点がある。

【0005】 上記リモートブートにおいて、各クライアントの機種が異なる場合には、機種ごとのOSプログラムファイルをサーバHD上に用意しておき、クライアントの機種に合ったOSプログラムファイルをリモートブート実行時に読み出す必要がある。

【0006】 この場合、サーバのHD上に置かれた複数のOSファイルと、クライアント機種との対応付けは、従来該サーバ及び該クライアントに装着するLANボードのMAC（Medium Access Control）アドレスを用いて行われていた。ここで、MACアドレスとはLANにおける通信データの授受を行う際に、各情報処理装置のLANボードのアドレスとして使われるものであり、該ボード一枚一枚に固有の値が割り当てられている。

【0007】 上記対応付けを以下具体的に説明する。

【0008】 図5に、リモートブート実行時のシステム構成を示す。クライアント2、クライアント3、クライアント4、クライアント5は、ネットワーク6を経由し、サーバ1のHD7上のファイル8を共通にアクセスする。クライアント2及びクライアント3はXタイプの機種であり、クライアント4及びクライアント5はYタイプの機種である。また、クライアント2、3、4、5のMACアドレスはそれぞれ5A003B77H（Hは16進数を示す）、4B931211H、2C984A67H、75682732Hである。

【0009】 図8に、前記各クライアントがリモートブートを実行する際に必要な、サーバHD7上のファイル構成を示す。IPL（Initial Program

Loader) 21、31はOSをロードするためのプログラムである。DOS. SYS22、32とINOUT. SYS23、33はOSのカーネルプログラムである。このうち、DOS. SYS22、32はファイル管理、メモリ管理、プロセス管理等のOSの基本機能を持ち、INOUT. SYS23、33は周辺装置を制御する。COM. COM24、34はコマンドインタプリタであり、入力したコマンドを解析し、指定されたコマンドプログラムを実行する。CONF. SYS25、35はユーザが固有の機能をOSに追加するため、デバイスドライバ等を指定する設定ファイルである。AUTO. BAT26、36はOSの起動時に自動的に実行するプログラム等の設定を行う自動実行バッチファイルである。なお、上述のファイルを総称してOSファイル群と呼ぶ。ディレクトリXTYPE81の下に格納されたOSファイル群は、前記Xタイプの機種用であり、ディレクトリYTYPE82の下に格納されたOSファイル群は、前記Yタイプの機種用である。MATCH. SYS83は、前記MACアドレスと前記OSファイル群との対応付けを行うためのファイルである。

【0010】図9は、テキストファイルである前記MATCH. SYS83の内容である。1行目、2行目は、クライアント2、クライアント3のMACアドレスである5A003B77H、4B931211Hと前記ディレクトリXTYPE81が対応付けられていることを示し、3行目、4行目は、クライアント4、クライアント5のMACアドレスである2C984A67H、75682732Hと前記ディレクトリYTYPE82が対応付けられていることを示している。

【0011】このように、クライアントは、該クライアントのLANボードのMACアドレスと、前記対応付けファイルMATCH. SYS83を用いて、該クライアント機種用のOSファイル群が格納されたディレクトリを判別し、該ディレクトリの下にOSファイル群を用いてリモートブートを行なう。これにより、クライアント機種に合ったOSファイル群を選択出来、複数のクライアント機種が混在した場合にも正しくリモートブートできる。

【0012】リモートブートについては、米国特許No. 240, 955に開示されている。

【0013】

【発明が解決しようとする課題】ところで、OS、かな漢字変換プログラム、グラフィカルユーザインタフェースプログラム等の種類や立ち上げ後の初期状態は、ユーザ毎の作業内容や好みによって異なる。また、同一ユーザでも行う作業内容によって、使用したいOS、かな漢字変換プログラム、グラフィカルユーザインタフェースプログラム等の種類や立ち上げ後の初期状態が異なる場合がある。

【0014】しかし、従来のリモートブート機能では、

同一クライアントを複数のユーザが使用した場合、ユーザ毎にOSの種類やOSの立ち上げ後の初期状態を決める環境設定内容を選択できないという問題があった。

【0015】また、従来のリモートブート機能では、OS立ち上げ時にユーザがOSの種類やOS立ち上げ後の初期状態を決める環境設定内容を選択できないという問題があった。

【0016】したがって、本発明の目的は、ユーザ毎に使用するクライアントのOSの種類、及び該OSの立ち上げ時の環境設定内容を選択可能なリモートブートシステムを提供することにある。

【0017】

【課題を解決するための手段】上記目的を達成するために、本発明は、サーバとクライアントから成る情報処理システムにおいて、ユーザ識別子に対応するOSファイル群を記述した対応付けファイルと、ユーザ識別子取得手段、前記ユーザ識別子と前記対応付けファイルを用いて、サーバHD上に置かれた複数のOSファイル群の中から所望のOSファイル群を読み出すOSファイル読み出し手段を設けた。

【0018】

【作用】ユーザ識別子取得手段は、画面表示等でユーザ識別子の入力を促し、ユーザがキーボード等から入力するユーザ識別子を取得する。OSファイル読みだし手段は、該ユーザ識別子と、ユーザ識別子とOSファイル群との対応付けを記述した対応付けファイルとを用いて、サーバHD上に置かれた複数のOSファイル群の中から所望のOSファイル群を選択し、クライアント上に読み出す。これによって、予め対応付けファイルでユーザ毎に指定したOSファイル群を使用することができ、ユーザ毎にOSの種類、及び該OSの立ち上げ時の環境設定を選択できる。

【0019】

【実施例】以下、本発明の一実施例を図を用いて説明する。本実施例におけるシステム構成は、従来例において図5で示したものと同一である。

【0020】サーバ1のHD7には、クライアント2、クライアント3、クライアント4、クライアント5の立ち上げに必要な、機種及びユーザ毎のOSファイル群と、前記クライアント2、クライアント3、クライアント4、クライアント5の機種識別子及びユーザ識別子と前記OSファイル群との対応付けを行う対応付けファイルが格納されている。

【0021】図4は、本発明の一実施例におけるクライアント2の構成例である。41はCPU、42は主記憶、43はROMである。ROM43には、情報処理装置の制御を行うBIOSプログラム等が格納される。44はキーボード45からのキー入力を制御するキーボードコントローラ、46はHD47への読み書きを制御するHDコントローラ、48はFDドライブ49に装着さ

れるFDへの読み書きを制御するFDコントローラ、50はディスプレイ51への表示データを制御するための表示コントローラ、55はCPU41、主記憶42、ROM43と各コントローラ間の信号の伝送を行うバスである。LANボード9はLANコントローラ52、LANボードBIOS ROM53、MACアドレスROM54から構成される。LANボードBIOS ROM53にはリモートブートを制御するプログラムが格納される。

【0022】図6は、前記クライアント2のメモリマップの一例を示す図である。システムプログラム61、FD BIOSプログラム62、表示BIOSプログラム63は、前記ROM43に格納されている。システムプログラム61内には、図6(b)に示すように、ハードウェアのチェックを行う初期診断プログラム66、FDD等の組み込み周辺装置用ROM BIOSプログラム67、記憶装置からIPLを読み出し実行するブートプログラム68がある。前記LANボード9のBIOS ROM53内に格納されるLANボードBIOS ROMプログラム64は、本実施例ではC000:0000H番地から配置されている。該LANボードBIOS ROMプログラム64の先頭アドレス部分には、図6(c)に示すように、特定のコード例えば55H、AAHと、前記コードに続いて初期化プログラム69が書き込まれる約束になっている。また、アドレスC000:1000H番地からは、割り込みによりサービスされるFDへの入出力を、サーバHD7への入出力に置き換えるためのFD BIOSエミュレータプログラム70が格納されている。また、0000:0000H番地から0000:03FFH番地の部分には割り込みベクタテーブル65が置かれている。

【0023】図11に、対応付けファイルMATCH、SYS105の内容を示す。MATCH、SYS105は、機種識別子であるMACアドレスとユーザ識別子を合わせた識別子と、前記ユーザ毎に用意したOSファイル群とを対応させるためのファイルである。本実施例では、ユーザ識別子としてSUZUKI、SATOを使用する。

【0024】例えば、1行目はクライアント2のMACアドレス5A003B77Hとユーザ識別子SUZUKIからOSファイル群が格納されたディレクトリとしてXSUZUKIを対応させ、2行目はクライアント2のMACアドレス5A003B77Hとユーザ識別子SATOからOSファイル群が格納されたディレクトリとしてXSATOを対応させることを示している。以下3行目から8行目まで同様の対応付けを示している。

【0025】図10に、図5のシステム構成において各クライアントがリモートブートを実行する際に必要な、サーバHD7上のファイル構成を示す。XSUZUKI101、YSUZUKI102、XSATO103、Y

SATO104は、クライアント機種及びユーザ毎に用意したディレクトリであり、各ディレクトリにOSファイル群が格納されている。以下でOSファイル群を構成する各ファイルについて説明するが、前記各ファイルの詳細な機能説明については、従来の技術で記した同一名ファイルの説明と同様であるのでここでは省略する。

【0026】IPL71、DOS、SYS72、INOUT、SYS73、COM、COM74は、Xタイプの機種に固有のOSファイルである。CONF、SYS75、AUTO、BAT76は、Xタイプの機種でSUZUKIがリモートブートを行う場合の固有の環境設定がなされているファイルである。同様に、CONF、SYS75'、AUTO、BAT76'は、Xタイプの機種でSATOがリモートブートを行う場合の固有の環境設定がなされているファイルである。IPL81、DOS、SYS82、INOUT、SYS83、COM、COM84は、Yタイプの機種に固有のOSファイルである。上述のOSファイルと名前は同じであるが、内容は異なっている。CONF、SYS85、AUTO、BAT86は、Yタイプの機種でSUZUKIがリモートブートを行う場合の固有の環境設定がなされているファイルである。同様に、CONF、SYS85'、AUTO、BAT86'は、Yタイプの機種でSATOがリモートブートを行う場合の固有の環境設定がなされているファイルである。

【0027】次に、リモートブート時の動作の説明をする前に、クライアント2がFDからブートを行った場合の動作について、図2のフローチャートを用いて以下で説明する。

【0028】ユーザにより電源投入あるいはリセット指示がかかると、上記システムプログラム61が実行される。システムプログラム61内の初期診断プログラム66は、ステップ201においてメモリチェック等の初期診断を行う。次に、前記周辺装置用ROM BIOSプログラム67に制御が移りステップ202からステップ206までの処理を実行する。ステップ202ではFDD49等の組み込み周辺機器、及びそのコントローラを初期化する。この時、FDD49等に対する入出力処理をサービスするためのROMプログラムのエントリ番地を、割り込みベクタとして主記憶42に設定する。FDD49等に対する入出力処理を行う際には、ソフトウェア割り込みをプログラムから発行することにより、前記割り込みベクタに指定された処理プログラムを呼び出すことが出来る。ステップ203では、情報処理装置のオプションスロット等に装着された拡張ボードのBIOS ROMの有無を調査する。本実施例では、拡張ボードBIOS ROMプログラムの先頭アドレスに位置するデータを55H、AAHと約束しているため、55H、AAHのデータが連続して存在するアドレスを調査する。該調査の結果、拡張ボードBIOSが検出されると、ス

ステップ204において該拡張ボードBIOSの初期化プログラムが実行される。後述するリモートブート時にはLANボードBIOSが検出された後、該初期化プログラム69が実行される。次に、ステップ206において全アドレスを調査したかどうかによって拡張ボードBIOS調査の終了判定を行い、終了するまでステップ203～205を繰り返す。拡張ボードBIOSの調査が終了した場合は制御が上記ブートプログラム68に移り、ステップ206において、IPLをFDD49から読みだし実行する。次に、前記IPLがステップ207においてOSのカーネルプログラムであるDOS. SYSとINOUT. SYSを読みだし実行する。具体的には、INOUT. SYSが周辺装置の初期化（周辺装置の接続状態のチェック、ドライバルーチンの初期化）を行った後、OSの本体であるDOS. SYSの初期化ルーチンを実行し、作業領域の確保、バッファ領域の確保等を行う。次に、ステップ208においてCONF. SYSを参照し、指定された周辺装置等のドライバプログラムをシステムへ組み込んだ後、ステップ209でコマンドインタプリタであるCOM. COMを読み出し実行する。COM. COMはステップ210において、自動実行バッチファイルであるAUTO. BATを読み出し実行する。

【0029】図6で示したように、リモートブートを制御するLANボードBIOSプログラム64の先頭アドレスに格納されているデータは、55H, AAHである。したがって、上記ステップ203で該LANボードBIOSが検出される。該LANボードBIOSの初期化プログラム69がステップ204で実行される。

【0030】次に、図1のフローチャートを用い、前記初期化プログラム69の動作を説明する。

【0031】まず、ステップ1001でLANコントローラ5を介しサーバ1との接続を確立する。次に、ステップ1002でLANボード9上のMACアドレスROM54にアクセスし、MACアドレスの読み込みを行う。次に、ステップ1003でディスプレイ51にユーザ識別子入力画面を表示する。次に、ステップ1004でキーボード45にユーザ識別子が入力されると、ステップ1005でサーバHD7上の対応付けファイルMATCH. SYS105を読み込む。ステップ1006で前記MACアドレスと前記ユーザ識別子に一致する行を該対応付けファイルMATCH. SYS105の中からサーチすることによって、該MACアドレスと該ユーザ識別子に対応するOSファイル群が格納されたディレクトリ名を得て、ステップ1007で該ディレクトリ名を記憶する。

【0032】例えば、MACアドレスが5A003B77H、上記ユーザ識別子がSUZUKIであった場合は、図11で示した対応付けファイルMATCH. SYS105の1行目からXSUZUKIのディレクトリを

記憶するのである。次に、ステップ1008で上記FDD49に対する入出力処理をサービスするための割り込みベクタをFD BIOSエミュレータプログラム70をコールするためのアドレスに書き換え、LANボードBIOS初期化プログラム69を終了する。

【0033】図7は、図6で示した割り込みベクタテーブル65を拡大したメモリマップである。図7において、割り込みベクタテーブル65内の0000:0064H番地は上述のFDD49に対する入出力処理サービスをコールするためのエントリ番地が格納されている領域である。上述のように図2のステップ202では、図7(a)に示すように前記エントリ番地をFD BIOSプログラム62が格納されているF000:0000H番地に設定している。上記LANボードBIOSプログラム64の初期化プログラム69は、上述のように図1のステップ1008において前記エントリ番地を図7(b)に記すように、FD BIOSエミュレータプログラム70が格納されているC000:0000番地に書き換える。

【0034】次に、上記FD BIOSエミュレータプログラム70の動作を、図3のフローチャートを用いて以下説明する。FD BIOSエミュレータプログラム70は、ステップ301において、上述のステップ1007で記憶済のOSファイル群格納ディレクトリ名を読み出す。次に、FD BIOSエミュレータプログラム70を呼び出したプログラムからパラメータとして渡されるファイル名に、前記OSファイル群格納ディレクトリ名を付加する。次に、ステップ303で、前記付加を行なったファイル名を用いてサーバHD1からファイルを読み出す。

【0035】例えば、上述のようにMACアドレスが5A003B77Hでユーザ識別子がSUZUKIであった場合は、XSUZUKIというディレクトリ名が記憶されているので、“IPL”というファイル名をパラメータとして読み出し要求があった場合には、“XSUZUKI/IPL”というファイル名でサーバHD1から読み出しを行なうのである。

【0036】上述の割り込みベクタ書き換えと、前記FD BIOSエミュレータプログラム70によって、図2のステップ206～210においてFDから読み出していたOSファイルは、サーバHD1上のMACアドレスとユーザ識別子の組合せ毎に用意したディレクトリの下に格納されたOSファイル群の中から読み出される。

【0037】以上の動作によって、ユーザ毎に異なるIPL、OSのカーネルプログラム、コマンドインタプリタ、及び自動実行バッチファイル、周辺装置制御ファイル等を使用してOSを立ち上げることができる。これにより、1台の情報処理装置を複数のユーザで交代に使う場合にも、各ユーザに最適な環境を提供出来るという効果がある。

【0038】前記実施例では、MACアドレスとユーザ識別子でOSファイル群を識別し、OSをブートする例を示したが、ROM43に書かれた機種コード等を用いてクライアントの機種を識別することも可能である。このようにクライアントの機種が特定出来れば、MACアドレス以外の識別子を用いても良い。また、同一機種のみでクライアントが構成される場合は、ユーザ識別子だけを用いても良い。さらに、ユーザ識別子に替えて、OSの種類を示すOS識別子を用いることによってリモートブートするOSの選択をすることも可能であり、環境設定ファイルの種類を示す環境識別子を用いることによって、リモートブート時に利用する環境設定ファイルを選択することも出来る。もちろん、各識別子を組み合わせて使用することも可能である。

【0039】

【発明の効果】以上説明したように、本発明によれば、リモートブートによって立ち上げるOSの種類、環境をユーザ毎に設定でき、1台の情報処理装置を複数のユーザで交代に使う場合にも各ユーザに最適な環境を提供出来るという効果がある。

【図面の簡単な説明】

【図1】本発明の一実施例におけるLANボードBIOSの初期化プログラムの実行内容を示すフローチャートである。

【図2】本発明の一実施例におけるクライアントの立ち上げ動作を示すフローチャートである。

【図3】本発明の一実施例におけるFD BIOSエミュレータプログラムの動作を示すフローチャートである。

【図4】本発明の一実施例におけるクライアントの構成図である。

【図5】従来例および本発明の一実施例におけるシステム全体の構成を示す図である。

【図6】本発明の一実施例におけるクライアントのメモリマップを示す図である。

【図7】本発明の一実施例におけるクライアントの割込みベクタを示す図である。

【図8】従来のリモートブートにおけるサーバのHD上のリモートブート用OSファイル構成を示す図である。

【図9】従来のリモートブートにおける対応付けファイルの内容を示す図である。

【図10】本発明の一実施例におけるサーバのHD上のリモートブート用OSファイル構成を示す図である。

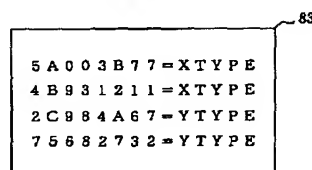
【図11】本発明の一実施例における対応付けファイルの内容を示す図である。

【符号の説明】

- 1…サーバ、
- 2・3・4・5…クライアント、
- 6…ネットワーク、
- 7…HD、
- 8…ファイル、
- 9…LANボード、
- 41…CPU、
- 42…主記憶、
- 43…ROM、
- 44…キーボードコントローラ、
- 45…キーボード、
- 46…HDコントローラ、
- 47…FD、
- 48…FDコントローラ、
- 49…FDD、
- 50…表示コントローラ、
- 51…ディスプレイ、
- 52…LANコントローラ、
- 53…LANボードBIOS ROM、
- 54…MACアドレスROM、
- 61…システムプログラム、
- 62…FD BIOSプログラム、
- 63…表示BIOSプログラム、
- 64…LANボードBIOSプログラム、
- 65…割込みベクタテーブル、
- 66…初期診断プログラム、
- 67…周辺装置用ROM BIOSプログラム、
- 68…ブートプログラム、
- 69…初期化プログラム、
- 70…FD BIOSエミュレータプログラム、
- 81・82・101・102・103・104…リモートブート用OSファイル格納ディレクトリ、
- 83・105…対応付けファイルの内容。

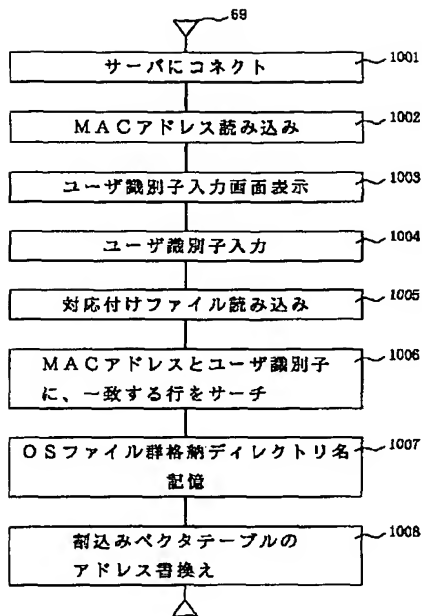
【図9】

図9



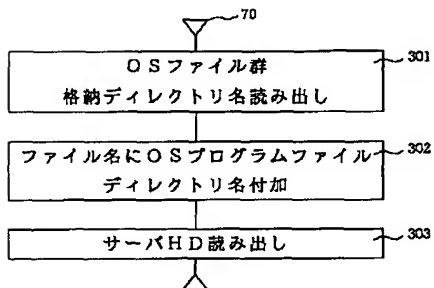
【図1】

図1



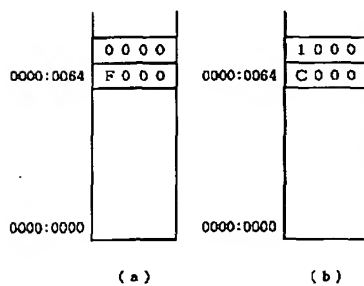
【図3】

図3



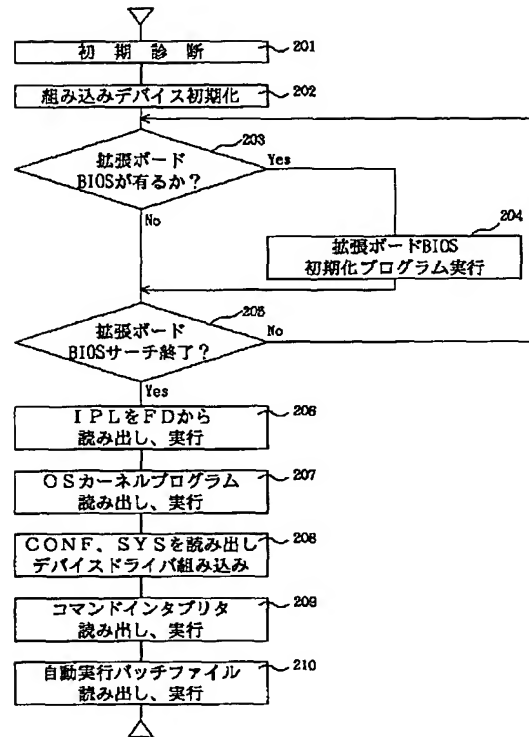
【図7】

図7



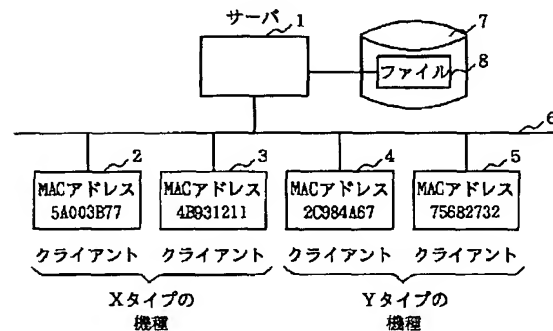
【図2】

図2



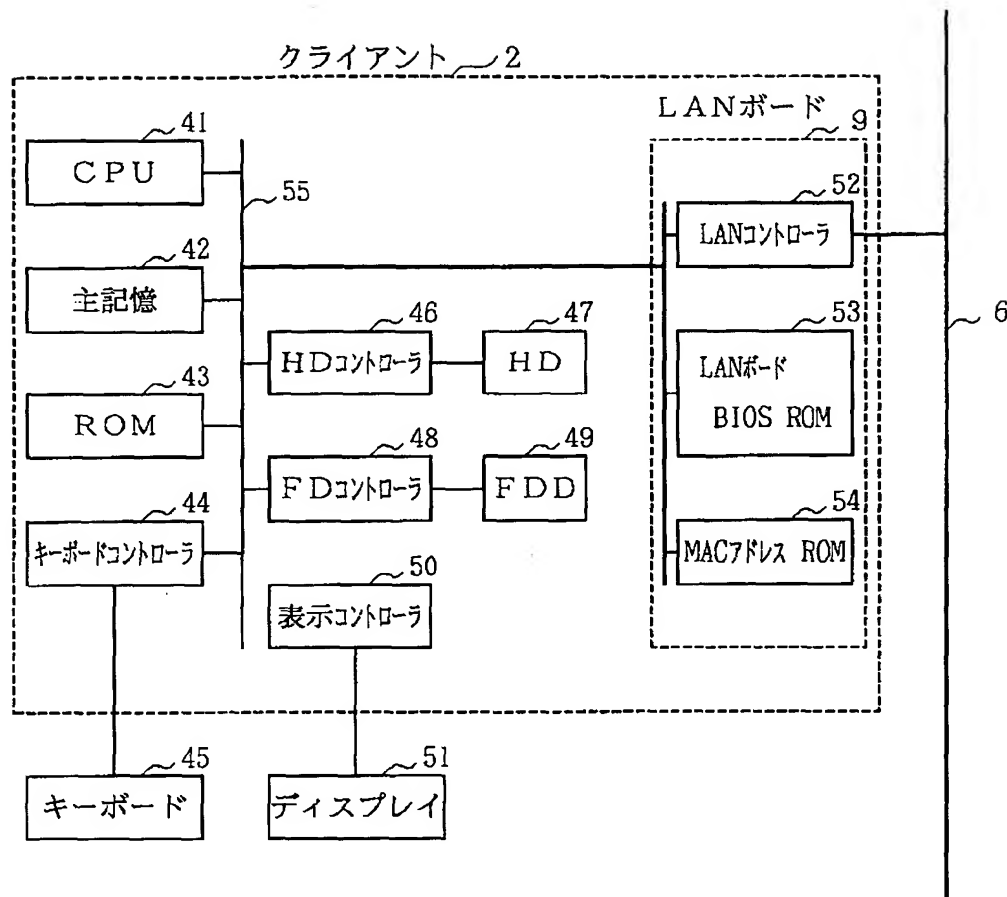
【図5】

図5



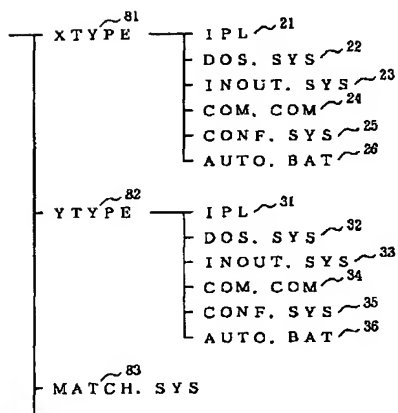
【図4】

図4



【図8】

図8



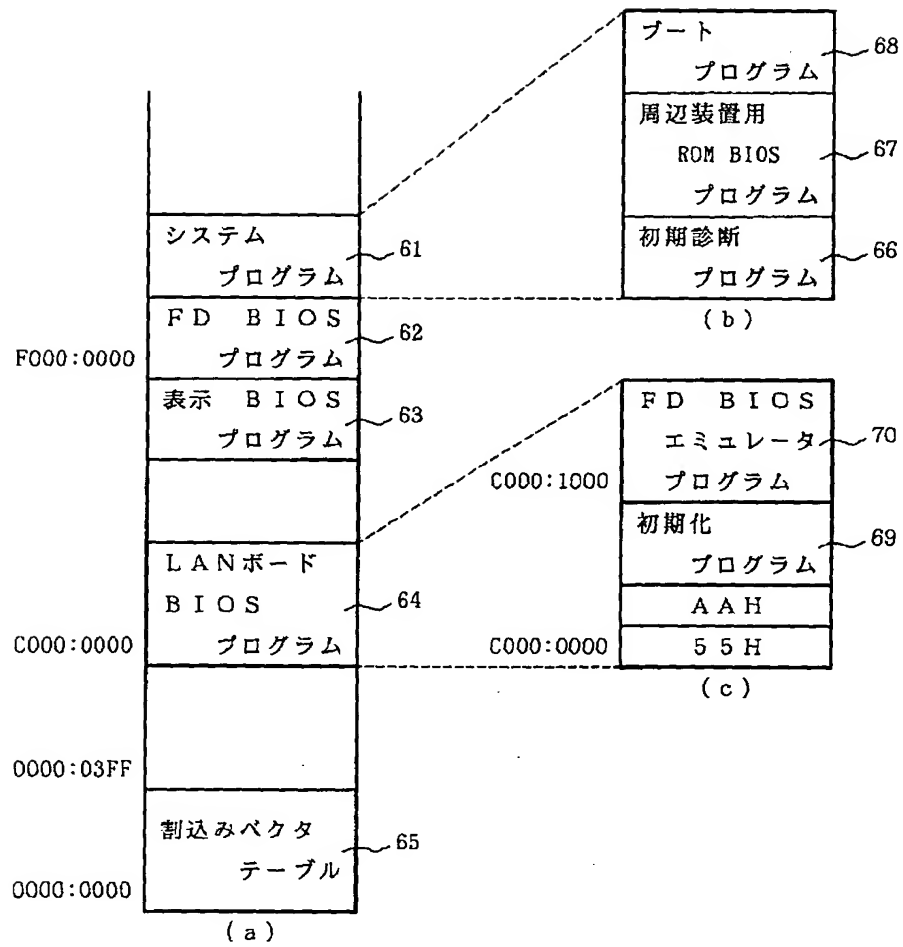
【図11】

図11

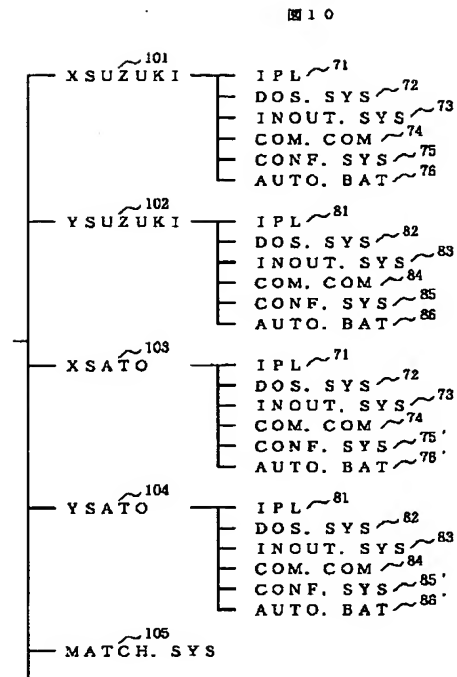
5A003B77, SUZUKI	= XSUZUKI
5A003B77, SATO	= XSATO
4B931211, SUZUKI	= XSUZUKI
4B931211, SATO	= XSATO
2C984A67, SUZUKI	= YSUZUKI
2C984A67, SATO	= YSATO
75682732, SUZUKI	= YSUZUKI
75682732, SATO	= YSATO

【図6】

図6



【図10】



フロントページの続き

(72)発明者 中根 啓一
 神奈川県横浜市戸塚区吉田町292番地株式
 会社日立製作所マイクロエレクトロニクス
 機器開発研究所内